Self-Supervised Correspondence and Co-Segmentation in Latent Diffusion Models

Rehan Ahmad

Department of Physics, LUMS

1 Introduction

Recent advancements in diffusion models have demonstrated the power of self-attention and cross-attention in uncovering meaningful object structures within images. Previous works have leveraged self-attention to perform segmentation by exploiting its ability to group spatially related regions without requiring explicit supervision [8]. Similarly, cross-attention maps have been utilized to establish local correspondences between images [6] and find keypoints in an unsupervised fashion [5].

These methods are training-free or require only minimal training as they directly exploit the attention maps of pre-trained diffusion models. However, they suffer from key deficiencies. While [8] introduces an efficient way to cluster Self-Attention using a grid-based approach, the method relies upon hyper-parameters such as the merging threshold and the number of points on the grid to obtain a natural clustering of meaningful objects in the image such as "eye", "paw" and "ears". Similarly, while [7] is able to systematically train an embedding to focus on an object of interest, the position that the embedding must focus on must be defined manually before training. More crucially, the method is limited to "local correspondences".

Given these constraints, we ask: *Is it possible to leverage the supervisory signals within self-attention to extract key features across a collection of objects—without manually defining positions or relying on hand-tuned hyperparameters*? Towards this aim, we perform feature matching between two images, I_s and I_t using SD2 to find semantically consistent regions across images without requiring explicit training. This follows the recent trend of using the over-parametrized pipeline of Diffusion Models for extracting useful representations [4] [9].

2 Method

Given a feature map f^l extracted from a U-Net layer , self-attention computes a similarity-based transformation:

$$A_{self} = \left(Q^l, K^l\right) = \text{softmax}\left(\frac{Q^l (K^l)^T}{\sqrt{d}}\right) \tag{2.1}$$

Where $Q \in \mathbb{R}^{H \times D \times N}$ represent the query matrix, H is the number of attention heads, D is the feature dimension per head, and N is the number of spatial tokens. Similarly, $K \in \mathbb{R}^{H \times D \times N}$ denote the key matrix. The dimensions of A_{self} are $A_{\text{self}} \in \mathbb{R}^{H \times N \times N}$. For our purposes, we would be averaging all the attention maps we are working with across heads:

$$A = \frac{1}{H}\sum_{h=1}^{H} A^{(h)} \in \mathbb{R}^{N\times N} \tag{2.2}$$

Where $N \in 64, 256, 1024, 4096$. For the proof of concept notebook, we are working with the middle layer of U-Net where the spatial dimension N equals 1024. The latter half of our attention matrices is partitioned into an 32×32 grid. Each grid cell corresponds to a submatrix of size 32×32 (since $1024/32 = 32$). Thus, A is partitioned as $A[i, j, x, y]$ where (i, j) are the pixels of interest and $(x, y) \in \mathbb{R}^{32\times 32}$ shows the associations of each pixel with the other. Each submatrix $A_{x,y}$ associated with (i, j) is normalized along its rows to ensure that it constitutes a valid probability distribution:

$$A_{i,j} = \frac{A_{i,j}}{\sum_{k=1}^{32} A_{i,j}(k, :)}, \quad A_{i,j} \in \mathbb{R}^{32\times 32}. \tag{2.3}$$

A central player of our method is **cross self-attention**. Also known as mutual self-attention or KV injection, cross-self-attention have been extensively used in context of SD2 to perform wide-array of tasks such as Style Transfer [3], Localized Edits [2] and Appearance Transfer [1]. In Cross Self-Attention features f^l_{ref} from a source image I_{source} interact with the features from a target image I_{target}. Thus, the usual Self-Attention matrix is replaced as following:

$$A_{cs} = \left(Q^l_{\text{source}}, K^l_{\text{target}}\right) = \text{softmax}\left(\frac{Q^l_{\text{source}} \left(K^l_{target}\right)^T}{\sqrt{d}}\right) \tag{2.4}$$

Where,

$$Q^l_{target} = W_Q f^l_{target} \quad K^l_{ref} = W_k f^l_{ref}$$

We define two types of cross self-attention maps: A cross-self attention matrix that highlights how the source queries attend to the target queries and another one that highlights how the target queries attend to source queries. After averaging across heads and normalization, these two maps can be written as thus, :

$$A_{\text{stg}} = \text{softmax}\left(\frac{Q_s \cdot K_t^T}{\sqrt{D}}\right), \quad A_{\text{stg}} \in \mathbb{R}^{N\times N} \tag{2.5}$$

$$A_{\text{tts}} = \text{softmax}\left(\frac{Q_t \cdot K_s^T}{\sqrt{D}}\right), \quad A_{\text{tts}} \in \mathbb{R}^{N\times N} \tag{2.6}$$

Our method is based on a principle similar to inverse consistency used in image registration. This metric quantifies the distance between the composition of mappings from one image to another. Along similar lines, we define an **inverse mapping score** which evaluates how well a transformation from a source domain to a target domain preserves spatial relationships when mapped back to the source domain. Given a mapping $f : S \to T$ that transforms a source coordinate x_s to a target coordinate x_t, the inverse mapping function $g : T \to S$ attempts to map x_t back to x'_s. The error in reconstruction is given by:

$$d(x_s, x'_s) = \|x_s - g(f(x_s))\| \tag{2.7}$$

where x_s is the original point in the **source domain**, $x_t = f(x_s)$ is the corresponding mapped point in the **target domain**, and $x'_s = g(x_t)$ is the reconstructed source point from the target mapping. The Inverse Mapping Score can be defined as:

$$IMS = \frac{1}{N}\sum_{i=1}^{N} d(x_s^i, x_s'^i) \tag{2.8}$$

Using cross-self attention, we define $d(x_s, x_s')$ as following:

$$d(x_s, x_s') = \|x_s - A_{tts}(A_{stt}(x_s))\| \tag{2.9}$$

Another quantity of interest is **Inverse mapping entropy**. This is used to quantify the uncertainty in the inverse mapping process. Given a probability distribution $P(S_i|T_j)$ that represents the likelihood of a source coordinate S_i mapping back from a target coordinate T_j, the Inverse Mapping Entropy is computed as:

$$H(T_j) = -\sum_i P(S_i|T_j)\log P(S_i|T_j)$$

where $P(S_i|T_j)$ is the probability of source location S_i being the correct inverse mapping of target location T_j. If a target point T_j maps back to a single, well-defined source S_i with high probability (i.e., $P(S_i|T_j) \approx 1$), the entropy is **low**, meaning the mapping is **deterministic**. If a target point T_j maps back to multiple sources S_i with nearly equal probabilities, the entropy is **high**, meaning the mapping is **uncertain and diffuse**.

If a single target receives too many mappings, it may be an **overloaded point** (a collapse error). To acocunt for this, we use mapping density. We simply count how many times a target appears.

$$MD(y) = \sum 1(Tgt \to Src = y)$$

where 1 is an indicator function that counts occurrences of y. High $MD(y)$ indicates too many inputs mapping to one point (possible issue) and low $MD(y)$ indicates that the target is sparsely mapped (more likely to be valid).

We can combine mapping density and entropy to compute NIMs.

$$NIMS(y) = \frac{IME(y)}{\log MD(y)}$$

wher $IME(y)$ captures the **diversity** of sources mapping to y and $MD(y)$ accounts for how **overloaded** the target is. A high $NIMS(y)$ indicates that many sources mapping evenly (good distribution) while a Low $NIMS(y)$ indicate many sources collapsing into one target (bad).

TO BE COMPLETED BY 15 TH FEBRUARY

References

[1] Yuval Alaluf, Daniel Garibi, Or Patashnik, Hadar Averbuch-Elor, and Daniel Cohen-Or. Cross-image attention for zero-shot appearance transfer. In ACM SIGGRAPH 2024 Conference Papers, pages 1–12, 2024.

[2] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 22560–22570, 2023.

[3] Sooyeon Go, Kyungmook Choi, Minjung Shin, and Youngjung Uh. Eye-for-an-eye: Appearance transfer with semantic correspondence in diffusion models. arXiv preprint arXiv:2406.07008, 2024.

[4] Yutong He, Naoki Murata, Chieh-Hsin Lai, Yuhta Takida, Toshimitsu Uesaka, Dongjun Kim, Wei-Hsiang Liao, Yuki Mitsufuji, J Zico Kolter, Ruslan Salakhutdinov, et al. Manifold preserving guided diffusion. arXiv preprint arXiv:2311.16424, 2023.

[5] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Xingzhe He, Hossam Isack, Abhishek Kar, Helge Rhodin, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised keypoints from pretrained diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 22820–22830, 2024.

[6] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised semantic correspondence using stable diffusion. Advances in Neural Information Processing Systems, 36, 2024.

[7] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised semantic correspondence using stable diffusion. Advances in Neural Information Processing Systems, 36, 2024.

[8] Junjiao Tian, Lavisha Aggarwal, Andrea Colaco, Zsolt Kira, and Mar Gonzalez-Franco. Diffuse attend and segment: Unsupervised zero-shot segmentation using stable diffusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3554–3563, 2024.

[9] Yingheng Wang, Yair Schiff, Aaron Gokaslan, Weishen Pan, Fei Wang, Christopher De Sa, and Volodymyr Kuleshov. Infodiffusion: Representation learning using information maximizing diffusion models. In International Conference on Machine Learning, pages 36336–36354. PMLR, 2023.